a) Sequential file allocation strategie:

#include<stdio.h>

void main()

{

int n,i,j,b[20],sb[20],t[20],x,c[20][20];

printf("\n enter no. of files");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("\n enter no. of blocks occupied by file %d :",i+1);

scanf("%d",&b[i]);

printf("\n enter the starting block of file %d :",i+1);

scanf("%d",&sb[i]);

t[i]=sb[i];

for(j=0;j<b[i];j++)

c[i][j]=sb[i]++;

}

printf("\n file name\t start block \t length\n");

for(i=0;i<n;i++)

printf("%d\t%d\t%d\n",i+1,t[i],b[i]);

}

[student@localhost]$ vi seq.c

[student@localhost ]$ gcc seq.c

[student@localhost]$ ./a.out

**Output:**

enter no. of files3

enter no. of blocks occupied by file 1 :4

enter the starting block of file 1 :2

enter no. of blocks occupied by file 2 :6

enter the starting block of file 2 :8

enter no. of blocks occupied by file 3 :6

enter the starting block of file 3 :12

file name start block length

1 2 4

2 8 6

3 12 6

b) Indexed file allocation strategies:

#include<stdio.h>

void main()

{

int n,m[20],i,j,sb[20],s[20],b[20][20],x;

printf("\n enter no. of files");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("\n enter index block & size of file %d:",i+1);

scanf("%d%d",&sb[i],&s[i]);

printf("\n enter blocks occupied by file %d:",i+1);

scanf("%d",&m[i]);

printf("\n enter blocks of file %d:",i+1);

for(j=0;j<m[i];j++)

scanf("%d",&b[i][j]);

}

printf("\n file \t index\t length\n");

for(i=0;i<n;i++)

{

printf("%d\t%d\t%d\n",i+1,sb[i],m[i]);

}

}

**Output:**

[student@localhost ]$ vi index1.c

[student@localhost]$ gcc index1.c

[student@localhost]$ ./a.out

enter no. of files2

enter index block & size of file 1:4

5

enter blocks occupied by file 1:1

enter blocks of file 1:2

enter index block & size of file 2:5

3

enter blocks occupied by file 2:3

enter blocks of file 2:2

3

4

file index length

1 4 1

2 5 3

c. Linked file allocation Strategies

#include<stdio.h>

struct file

{

char name[10];

int start,size,block[10];

}f[10];

void main()

{

int i,j,n;

printf("\n enter no. of files");

scanf("%d",&n);

for(i=0;i<n;i++)

{

printf("\n enter file name:");

scanf("%s",f[i].name);

printf("\n enter starting block:");

scanf("%d",&f[i].start);

f[i].block[0]=f[i].start;

printf("\n enter no. of blocks:");

scanf("%d",&f[i].size);

printf("\n enter block numbers");

for(j=1;j<=f[i].size;j++)

{

scanf("%d",&f[i].block[j]);

}

}

printf("\n file\t start \t size\t block \n");

for(i=0;i<n;i++)

{

printf("%s \t %d\t%d\t",f[i].name,f[i].start,f[i].size);

for(j=1;j<=f[i].size-1;j++)

printf("%d---->",f[i].block[j]);

printf("%d",f[i].block[j]);

printf("\n");

}

}

**Output:**

[student@localhost]$ vi linked1.c

[student@localhost]$ gcc linked1.c

[student@localhost]$ ./a.out

enter no. of files2

enter file name:document1

enter starting block:3

enter no. of blocks:5

enter block numbers1

4

9

7

5

enter file name:document2

enter starting block:5

enter no. of blocks:4

enter block numbers6

4

7

9

file start size block

document1 3 5 1---->4---->9---->7---->5

document2 5 4 6---->4---->7---->9